Optical Switching and Networking 15 (2015) 121–133

Contents lists available at ScienceDirect

Optical Switching and Networking

journal homepage: www.elsevier.com/locate/osn

Energy-efficient Dynamic Bandwidth Allocation for EPON networks with sleep mode ONUs

Chengjun Li [a], Wei Guo [a,*], Weisheng Hu [a], Ming Xia [b]

[a] State Key Lab of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai, 200240, China
[b] Erisson Research, USA

ARTICLE INFO

Article history:
Received 4 March 2014
Received in revised form
21 July 2014
Accepted 31 July 2014
Available online 8 August 2014

Keywords:
Ethernet Passive Optical Network (EPON)
Medium Access Control (MAC)
Dynamic Bandwidth Allocation (DBA)
Energy efficiency
Sleep mode ONU

ABSTRACT

Putting no-load Optical Network Units (ONUs) into sleep mode has been recognized as a promising solution to achieve high energy efficiency for Ethernet Passive Optical Networks (EPONs). To minimize ONU active time, the upstream and downstream transmission windows of each ONU should be overlapped as much as possible. However, the existing Dynamic Bandwidth Allocation (DBA) schemes for EPONs only consider the upstream bandwidth requests. In this paper, we design a novel energy-efficient DBA framework to reduce energy consumption, and meanwhile to guarantee network performance based on bandwidth requests in both downstream and upstream directions. Our DBA framework includes an energy-efficient MAC control scheme, a grant sizing policy and two grant scheduling algorithms. With the MAC control scheme, the Optical Line Terminal (OLT) allocates transmission windows to all ONUs and puts them into low power mode (sleep or doze mode) in each polling cycle. We formulate the energy efficient grant scheduling problem and introduce two energy-efficient grant scheduling algorithms to solve this problem. Results show the effectiveness of the proposed framework and highlight the merits of our algorithms.

1. Introduction

In recent years, energy efficiency of telecommunication networks has caught the world's attention [1,2]. Compared with metro and core networks, access networks consume a major part of the total energy consumed by the IT infrastructure [3]. With the increasing demands for broadband access, the number of subscribers and data rates are expected to significantly increase in near future. Thus, the explosive growth of access networks results in huge power consumption, which makes energy saving technologies desirable [4–7].

Ethernet Passive Optical Network (EPON) has been widely deployed in recent years as an important access technology. An EPON comprises one OLT at the Central Office (CO), which connects multiple ONUs in a tree structure through optical fiber links. The OLT broadcasts downstream data to its ONUs, while ONUs transmit upstream traffic to the OLT in dedicated time slots. As all ONUs share the single upstream link to the OLT, the average upstream link utilization of each ONU is only about 1–5% [8]. Therefore, putting an ONU into sleep mode when there is no traffic for it is an attractive method to reduce power consumption [9–12]. This is expected to reduce the total ONU energy consumption by almost 80% [13,14]. Some ONU architectures have been proposed to allow ONUs to switch between active mode and sleep mode frequently without wasting much overhead time [15].

* Corresponding author.
E-mail addresses: lichengjun@sjtu.edu.cn (C. Li), wguo@sjtu.edu.cn (W. Guo).

http://dx.doi.org/10.1016/j.osn.2014.07.003
1573-4277/© 2014 Published by Elsevier B.V.

During a sleep period, the ONU turns off its transceivers in order to reduce energy consumption. To avoid packet loss, the incoming downstream and upstream data should be buffered at the OLT and the ONU, respectively. Furthermore, the ONU should wake up before the next communication with the OLT. Due to the bursty nature of traffic in access networks, the upcoming traffic from the OLT cannot be precisely predicted by the sleeping ONUs in advance. Therefore, the sleep periods should be carefully planned. The OLT should satisfy the bandwidth requests of both directions in each polling cycle to guarantee the network performance. And, on the other hand, it is preferred to let both upstream and downstream transmission windows overlap as much as possible, which can maximize the sleep time of ONUs. However, most existing Dynamic Bandwidth Allocation (DBA) schemes for conventional EPONs only consider the resource sharing in the upstream direction [16–20]. Without the sleep function in ONUs, the OLT always broadcasts the downstream traffic to them. It is not suitable for energy efficient EPONs with sleep mode ONUs because the downstream traffic would experience some extra delay due to the granted transmission windows based on the upstream requests only. Hence, it is desirable to investigate an energy efficient DBA framework to schedule the ONU transmission windows in both directions and achieve high energy efficiency with minimum network performance degradation.

In this work, we propose an energy-efficient MAC control scheme to control the operating mode of ONUs. The MAC control scheme facilitates collecting global knowledge of all upstream and downstream bandwidth requests, and schedules transmission slots for each ONU in an offline manner [19,20]. The grant sizing policy ensures the bandwidth requests of light-load ONUs and allocates transmission windows for heavy-load ONUs with excess bandwidth resource [21–23]. Then, we formulate the energy efficient grant scheduling problem and propose two energy efficient grant scheduling algorithms, the dynamic programming algorithm and the most fit first algorithm, to solve the problem. The dynamic programming algorithm obtains the global optimal polling order of the problem with exponential computational complexity. The most fit first algorithm obtains a local optimal polling order with polynomial computational complexity. Some extensions for extreme cases, such as light load case and asymmetric load case, are also introduced to improve the performance of our proposed algorithms. Finally, we investigate the performance of the proposed algorithms compared with the existing algorithms in different network scenarios.

The rest of the paper is organized as follows. Section 2 discusses the related literature. Section 3 presents the system model including the energy efficient MAC control scheme, the grant sizing policy and the energy efficient grant scheduling problem formulation. Section 4 introduces the two proposed algorithms. Section 5 introduces some extensions for special network cases. Section 6 conducts simulations and discusses the results. Finally, Section 7 concludes the paper.

2. Related work

In this section, we review the related works using software-based energy saving solutions.

The authors in [14] proposed a new green bandwidth allocation (GBA) framework. In GBA, the ONU goes into sleep mode for a certain time before waking up to send/receive a batch of buffered upstream/downstream traffic. Based on the maximum delay requirement rather than the availability of upstream and/or downstream traffics, the OLT calculates the allowable sleep time for each ONU.

The authors in [24] proposed two scheduling algorithms based on Energy Management Mechanism (EMM). In the upstream centric scheme (UCS), the OLT buffers the downstream traffic for a particular ONU until an upstream time slot is allocated for it. In the downstream centric scheme (DCS), the ONU has to wake up whenever there is upstream and downstream traffics for it. While it can achieve higher energy efficiency, UCS brings in additional delay and ignores the ONU downstream transmission fairness. Otherwise, with lower energy efficiency, DCS can guarantee the transmission fairness and obtain less downstream delay by waking ONUs up every time a transmission request arrivals.

The authors in [25] proposed an energy-saving scheme, named ESPS, to minimize the power consumption in EPONs. ESPS considers the bandwidth requests in both directions to get low packet delay. However, as an offline scheduling scheme, ESPS allocates the ONUs in a Round Robin manner, which does not exploit the global knowledge of all bandwidth requests. Then, a hybrid sleep mode is introduced as a combination of the ONU deep sleep mode and the independent sleep modes of the transmitter and the receiver. However, the implementation of this scheme cannot be easily derived from this work.

The authors in [26] proposed two just-in-time (JIT) dynamic bandwidth allocation algorithms (DBAs) based on the vertical-cavity surface-emitting laser (VCSEL) ONUs. In the first algorithm, JIT DBA with varying polling cycle times, the OLT waits until it receives REPORT messages from all ONUs and then calculates the allowable bandwidth for them. Based on the allocated bandwidth, the OLT selects the suitable mode (active, doze and sleep modes) for each ONU. The second algorithm, JIT DBA with fixed polling cycle times, can overcome the shortfall of short polling cycle times of the first algorithm through implementing fixed polling cycle times. However, in each DBA cycle, the OLT allocates the same bandwidth for all ONUs whose bandwidth requests may be different, resulting in some waste of bandwidth resource.

The authors in [27] proposed an Adaptive Delay-Aware Energy-Efficient (ADAEE) TDM-PON solution. ADAEE aims at saving as much energy as possible while meeting the PON access delay restrictions. A novel ONU architecture which is capable of selecting the suitable sleep mode and a novel algorithm to calculate the minimum and maximum sleep intervals for ONUs are presented. With the delay restrictions, the OLT calculates the sleep time for each ONU, and then selects the suitable sleep mode (light sleep mode or deep sleep mode). Detailed performance evaluation with different types of network traffics has been presented.

The authors in [28] proposed four sleep statuses of ONUs and two sleep mode scenarios. The four sleep statuses are achieved by the different transmitter and receiver states. Furthermore, in the first scenario, an ONU

is allowed to sleep for more than one cycle based on a predefined threshold. When the sleep time expires, the ONU wakes up and checks whether some downstream traffic is available. If there is no traffic for it, it computes the new sleep time and enters into sleep mode again. In the second scenario, the OLT schedules the downstream traffic of ONUs one by one. With the estimation of the traffic of other ONUs, the OLT determines the sleep time for the particular ONU within each DBA cycle.

The authors in [29] presented the experimental evaluation of an energy-efficient TDMA PON utilizing a cyclic sleep technique. And, two approaches for ONU sleep triggering are evaluated. In *downstream* (*DS*)-*based* triggering, the ONU enters into sleep mode based on the DS traffic only. And in the *Cooperative* triggering, the ONU enters into sleep mode when the traffic loads of both directions are low. To determine whether the traffic load is low or not, two estimated frame inter-arrival times are compared with two predefined thresholds for DS and US traffics, respectively. Both sleep triggering methods can achieve significant energy efficiency. However, only one ONU is considered in the system.

The authors in [30] proposed a sleep aware dynamic bandwidth allocation (SDBA) scheme to maximize energy efficiency while guaranteeing the QoS requirements on both upstream and downstream transmissions. Based on the strictest QoS requirements of all the traffic, the SDBA maximizes the polling cycle length to extend the ONU sleep time. And, an FPGA-based design and evaluation of the 10G-EPON systems are thoroughly described. Experimental results show that the SDBA can achieve significant energy saving while satisfying the strictest QoS requirements.

In our work, we investigate an offline-based energy efficient DBA framework in which the OLT schedules both upstream and downstream transmission windows of each ONU in each polling cycle. In the previous literature, the upstream and downstream transmission windows are forced to overlap entirely, which would bring about some extra delay. To avoid the extra delay, we permit the differences of time domain between the upstream and downstream transmission windows of the ONU. Compared to the entirely overlapping transmission windows, it may make the ONU stay longer in active mode. Therefore, we introduce two grant scheduling algorithms to overlap the upstream and downstream transmission windows as much as possible. Our proposed algorithms can achieve high energy efficiency with tolerable network performance degradation.

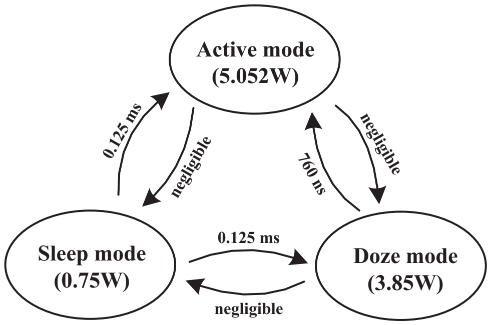

Fig. 1. ONU state transition diagram [13,15,26].

3. System model

3.1. Energy-efficient MAC control scheme

In conventional EPONs, the OLT broadcasts downstream traffic to all ONUs. Each ONU checks all the downstream traffic and receives the packets which are destined to it. So the ONU needs to stay awake all the time to avoid missing its downstream traffic. To reduce the power consumption, ONUs can enter into sleep mode when there is no traffic for them. The OLT grants both downstream and upstream transmission windows for each ONU, and the ONU wakes up right before the granted transmission windows to communicate with the OLT. So some modifications of the GATE message (defined in MPCP [31]) are needed to realize the sleep control function. The modified GATE message here should contain the information of both upstream and downstream transmission windows, which may not be the same in time domain. Based on the MPCP frame structure, the modified GATE message can be defined with a different opcode from the normal GATE message. One grant in the modified GATE message composes the granted upstream and downstream transmission windows. The REPORT message in our MAC control scheme is similar to that in the standard MPCP. ONUs only need to report the information of the queued data to the OLT within the REPORT message, as they do in conventional EPONs.

The proposed MAC control scheme operates based on three operating modes of ONUs: *sleep*, *active* and *doze*. In *sleep mode*, the ONU turns off both the transmitter and the receiver during a preset sleep time, and then wakes up. During the sleep time, no upstream or downstream traffic occurs in the ONU. In *active mode*, the ONU is powered on, and traffic is sent between the OLT and the ONU. In *doze mode*, the ONU turns off only the transmitter and keeps the receiver on to receive the downstream traffic and control messages from the OLT. Fig. 1 presents the power consumption of different operating modes and the state transition overhead time of the ONU proposed in [15]. With the specially designed receiver architecture, the ONU can regain synchronization from sleep mode to active mode with 0.125 ms overhead time while the overhead time from doze to active is only 760 ns. And our framework mainly focuses on this ONU architecture.

Our MAC control scheme is illustrated in Fig. 2. For a particular ONU, it experiences four phases in each polling cycle:

(1) At the beginning of the polling cycle, the ONU is in doze mode to receive the GATE messages from the OLT. The GATE message is used to inform the ONU of its allocated transmission window sizes and transmission begin times in both directions.
(2) After receiving the GATE message, the ONU is informed with its granted transmission windows and then enters into sleep mode or remains in doze mode.

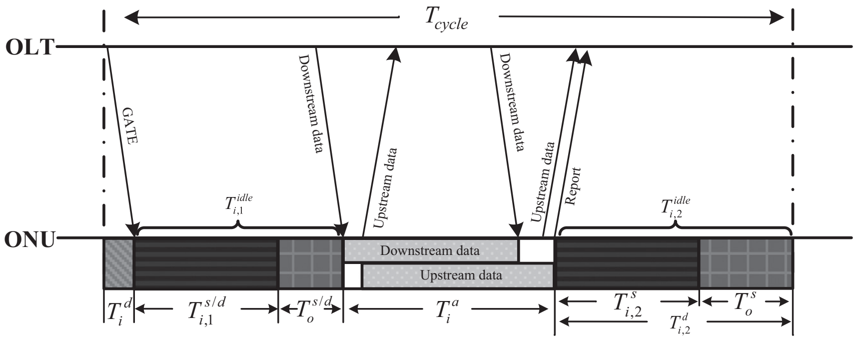

Fig. 2. Illustration of energy-efficient MAC control scheme.

The operating mode is calculated by the OLT based on the idle time between the GATE message receipt and the transmission start time.

(3) The ONU wakes up when its allocated transmission windows come and completes the clock synchronization with the OLT. After that, the OLT and the ONU transmit traffic to each other. When the upstream traffic ends, the ONU sends the REPORT message with the information of the queued data.

(4) Finally, after both upstream and downstream data transmissions finish, the ONU enters into sleep or doze mode until the next polling cycle begins.

To minimize the power consumption, the OLT decides the operating mode (sleep or doze mode) of each ONU during the idle time. $E^{idle,s}_{i,1}$ and $E^{idle,d}_{i,1}$ denote the energy consumed during $T^{idle}_{i,1}$ by ONU i in sleep mode and doze mode, respectively. For the first polled ONU in the polling cycle, the idle time before its active time slot $T^{idle}_{i,1}$ would be less than T^s_o, the overhead time from sleep mode to active mode. As the overhead time from doze mode to active mode T^d_o is only 760 ns, we can assume that $T^{idle}_{i,1}$ is always larger than T^d_o. So the ONU must remain in doze mode because the idle time is not enough for it to wake up from sleep mode to communicate with the OLT. Otherwise, we can find

$$T^{idle}_{i,1} = T^s_{i,1} + T^s_o = T^d_{i,1} + T^d_o \tag{1}$$

Thus, $E^{idle,s}_{i,1}$ and $E^{idle,d}_{i,1}$ can be formulated as follows:

$$E^{idle,s}_{i,1} = T^s_o P^s_o + (T^{idle}_{i,1} - T^s_o)P_s \tag{2}$$

$$E^{idle,d}_{i,1} = T^d_o P^d_o + (T^{idle}_{i,1} - T^d_o)P_d \tag{3}$$

If $E^{idle,s}_{i,1} < E^{idle,d}_{i,1}$, the OLT dictates the ONU to enter into sleep mode. Otherwise, the ONU remains in doze mode. Same as $T^{idle}_{i,1}$, the OLT lets the ONU enter the mode which consumes less energy during $T^{idle}_{i,2}$. Only one difference needs to be noted. As illustrated in Fig. 2, if the ONU dozes during $T^{idle}_{i,2}$, there is no state transition overhead time in the end of the idle time. It is because that the ONU remains in doze mode at the beginning of the next polling cycle.

Furthermore, ONUs with the sleep function can coexist with the legacy ONUs in the same network. In the upstream direction, the OLT allocates the upstream transmission window for the legacy ONUs with normal GATE, while in the downstream direction, the legacy ONUs are always active to receive all downstream data and only extracts the frames that are destined for it. The modified GATE messages would only be extracted by ONUs with the sleep function. Thus, this feature supports a smooth migration to the energy-efficient EPON without replacing all legacy ONUs at one time.

3.2. Grant sizing policy

Since our MAC control scheme is offline based, the OLT decides the allocated bandwidth for each ONU in upstream and downstream directions before each polling cycle and then determines the polling orders of all ONUs to minimize the total energy consumption. To increase the bandwidth utilization and improve the network performance, we make the grant sizing with the excess bandwidth distribution [21–23]. We denote the pre-defined polling cycle by T_{CYC}, which is the time during which all ONUs transmit/receive data to/from the OLT. In case of no service-level agreement (SLA) classification per ONU, the minimum guaranteed bandwidth for each ONU is computed as follows:

$$B_{MIN} = \frac{(T_{CYC} - N \cdot T_g)R}{N} \tag{4}$$

Let R^1_i and R^2_i denote the requested upstream and downstream bandwidths of ONU i, respectively. Without loss of generality, we consider the grant sizing in the upstream direction.

The OLT divides ONUs into two groups: under-loaded ONUs $i \in \mathbf{U}$ with $R^1_i \le B_{MIN}$, and over-loaded ONUs $i \in \mathbf{O}$ with $R^1_i > B_{MIN}$. For the under-loaded ONU, the granted bandwidth G^1_i is equal to the requested bandwidth R^1_i. The excessive bandwidth can be reserved for the over-loaded ONUs. The total excessive bandwidth can be computed as follows:

$$B^{total}_{excess} = \sum_{i \in \mathbf{U}} (B_{MIN} - R^1_i) \tag{5}$$

For each over-loaded ONU, the average excessive bandwidth is

$$B_{excess} = B_{excess}^{total}/M \tag{6}$$

where M denotes the number of elements in **O**. To improve the experienced delay for the over-loaded ONUs, the OLT grants excess bandwidth to them in an ascending order by their requested bandwidth. The granted bandwidth of over-loaded ONU i can be calculated as follows:

$$G_i^1 = \begin{cases} R_i^1 & \text{if } R_i^1 \le B_{MIN} + B_{excess}, \\ B_{MIN} + B_{excess} & \text{if } R_i^1 > B_{MIN} + B_{excess}. \end{cases} \tag{7}$$

Then the over-loaded ONU i is removed from **O**. The value of total excessive bandwidth B_{total}^{excess} is updated every time G_i^1 is assigned:

$$B_{excess}^{total} = B_{excess}^{total} - G_i^1 + B_{MIN} \tag{8}$$

The value of the average excessive bandwidth B_{excess} is also updated by Eq. (6). Using the same method, the OLT grants bandwidth to each ONU in the downstream direction.

3.3. *Problem formulation for energy-efficient grant scheduling*

In offline-based grant scheduling policies, the average packet delay is mainly related to the idle time within the polling cycle. Many existing offline grant scheduling policies [19–21] have been proposed to minimize the idle time to obtain less average packet delay. For the energy-efficient grant scheduling policy, extra idle times would be added in due to the asymmetric bandwidth requests of the two different directions. To reduce the impact on the network performance, we assume that there is no idle time between two adjacent ONU transmission windows in each direction. So the upstream and downstream transmission windows of one ONU cannot be overlapped entirely. If the previous ONU ends its transmission in one direction, the OLT can grant the transmission window to the next ONU without waiting for the transmission in the other direction. With this restriction, the OLT analyzes the granted bandwidth of all ONUs and determines how the multiple ONU transmission windows are ordered during the polling cycle to minimize the total energy consumption.

Formally, the problem of the energy efficient grant scheduling can be state as follows. The notations used hereafter are summarized in Table 1.

Given

(a) A set of N ONUs.
(b) Transmission direction ℓ, where $\ell=1$ means upstream, $\ell=2$ means downstream.
(c) The allocated bandwidth requests of ONUs G_i^ℓ; i for the ith ONUs.

Define

(a) $T_i^{\ell,s}$ denotes the transmission start time of ONU i in direction ℓ, and $T_i^{\ell,e}$ denotes the end time.
(b) T_i^a denotes the active time of ONU i. As the active time of ONU is the union of the upstream and downstream transmission durations, the formulation is stated as follows:

$$T_i^a = \max(T_i^{1,e}, T_i^{2,e}) - \min(T_i^{1,s}, T_i^{2,s}) \tag{9}$$

(c) E_i denotes the energy consumed during a polling cycle. It can be expressed as follows:

$$E_i = E_i^d + E_{i,1}^{idle} + E_i^a + E_{i,2}^{idle} \tag{10}$$

where the notations have been noted in Table 1.

Table 1
Definition of notations.

Notation	Description	Unit
T_{cycle}	Polling cycle length	s
E_i	Energy consumed by ONU i in T_{cycle}	J
T_i^d	Doze time in the front of polling cycle	s
E_i^d	Energy consumed in T_i^d	J
$T_{i,1}^s$	Sleep time before active time slot	s
$T_{i,1}^d$	Doze time before active time slot	s
$T_{i,1}^{idle}$	Idle time before active time slot	s
$E_{i,1}^{idle}$	Energy consumed in $T_{i,1}^{idle}$	J
T_i^a	Active time of ONU i	s
E_i^a	Energy consumed in T_i^a	J
$T_{i,2}^s$	Sleep time after active time slot	s
$T_{i,2}^d$	Doze time after active time slot	s
$T_{i,2}^{idle}$	Idle time after active time slot	s
$E_{i,2}^{idle}$	Energy consumed in $T_{i,2}^{idle}$	J
T_o^s	Overhead time (sleep to active/doze)	s
T_o^d	Overhead time (doze to active)	s
P_a	Power consumption in active mode	W
P_d	Power consumption in doze mode	W
P_s	Power consumption in sleep mode	W
P_o^s	Power consumption during T_o^s	W
P_o^d	Power consumption during T_o^d	W
R	Transmission speed	bit/s
N	Total number of ONUs	ONU
T_g	Guard time	s
N	The set of all N ONUs	

Objective: Find the ONU polling order σ ($[n]$ represents the nth ONU in the order) to minimize $\sum_{n=1}^{N} E_{[n]}$. With no idle time between two adjacent ONU transmission windows, the polling cycle length is fixed since the upstream and downstream bandwidths granted to each ONU are decided. As the ONU in sleep/doze mode consumes less energy than in active mode, less active time leads to more sleep time, which results in less total energy consumption. Thus, the objective can be simplified to find the ONU polling order σ to minimize the sum of $\sum_{n=1}^{N} T_{[n]}^a$.

Subject to

(a) All requests are allocated with sufficient time to be transmitted. So

$$G_i^\ell = (T_i^{\ell,e} - T_i^{\ell,s})R \tag{11}$$

(b) To guarantee the network performance in terms of the average packet delay, we make that the upstream/downstream traffic of one ONU can be sent immediately after

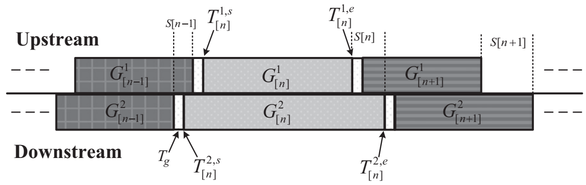

Fig. 3. Illustration of energy efficient grant scheduling problem.

the upstream/downstream transmission of the previously polled ONU ends. It means that there is no idle time between two adjacent ONU transmission windows in each direction. The formulation can be shown as follows:

$$T^{\ell,s}_{[n]} = T^{\ell,e}_{[n-1]} + T_g, \quad \forall n (1 \le n \le N) \tag{12}$$

Lemma III.1. *Given the ONU polling order σ and the allocated upstream and downstream transmission windows, the total active time of all ONUs during a polling cycle can be expressed as follows:*

$$\sum_{n=1}^{N} T^a_{[n]} = \left(\sum_{n=1}^{N} P_{[n]}\right) \Big/ R + \sum_{n=1}^{N} S_{[n]} \tag{13}$$

where $P_{[n]}$ and $S_{[n]}$ denote the granted bandwidth in the light-load direction of the nth ONU polled in σ and the absolute difference between the upstream and downstream transmission end times of the nth polled ONU, respectively. Fig. 3 *illustrates the energy efficient grant scheduling problem.*

Proof. From Fig. 3, we can find

$$S_{[n]} = |T^{1,e}_{[n]} - T^{2,e}_{[n]}| = \max(T^{1,e}_{[n]}, T^{2,e}_{[n]}) - \min(T^{1,e}_{[n]}, T^{2,e}_{[n]}) \tag{14}$$

Then, with Eqs. (9), (12), and (14), $\sum_{n=1}^{N}(T^a_{[n]})$ can be expressed as follows:

$$\begin{aligned}
\sum_{n=1}^{N} T^a_{[n]} &= \sum_{n=1}^{N} (\max(T^{1,e}_{[n]}, T^{2,e}_{[n]}) - \min(T^{1,s}_{[n]}, T^{2,s}_{[n]})) \\
&= \sum_{n=1}^{N} (\max(T^{1,e}_{[n]}, T^{2,e}_{[n]})) - \sum_{n=1}^{N-1} (\min(T^{1,e}_{[n]}, T^{2,e}_{[n]}) + T_g) \\
&= \sum_{n=1}^{N-1} S_{[n]} + \max(T^{1,e}_{[N]}, T^{2,e}_{[N]}) - (N-1)T_g \\
&= \sum_{n=1}^{N-1} S_{[n]} + S_{[N]} + \min(T^{1,e}_{[N]}, T^{2,e}_{[N]}) - (N-1)T_g
\end{aligned} \tag{15}$$

where $\min(T^{1,s}_{[n]}, T^{2,s}_{[n]})$ is the end time of the last polled ONU in the light-load direction. As there is no idle time between two adjacent ONU transmission windows, the sum of the granted transmission slots in the light-load direction $(\sum_{n=1}^{N} P_{[n]})/R$ equals $\min(T^{1,s}_{[n]}, T^{2,s}_{[n]})$ minus the total guard times $(N-1)T_g$. So Eq. (15) can be expressed as follows:

$$\sum_{n=1}^{N} T^a_{[n]} = \sum_{n=1}^{N} S_{[n]} + \left(\sum_{n=1}^{N} P_{[n]}\right) \Big/ R \tag{16}$$

As $\sum_{n=1}^{N} P_{[n]}$ is fixed in arbitrary polling orders, the energy efficient grant scheduling problem can be simplified to how to minimize $\sum_{n=1}^{N} S_{[n]}$. In the next section, we propose two algorithms to solve this problem.

4. Energy efficient grant scheduling algorithms

4.1. Dynamic programming algorithm

Let d_i denote the difference between the upstream and downstream granted transmission times of ONU *i*. It is given by

$$d_i = (G^1_i - G^2_i)/R. \tag{17}$$

For a given polling order σ, $S_{[n]}$ is calculated as follows:

$$S_{[n]} = \left| \sum_{j=1}^{n} d_{[j]} \right|. \tag{18}$$

The problem can be viewed as a discrete multistage decision process to facilitate the adoption of dynamic programming. The number of stages is equal to *N*. The pseudocode is given in Algorithm 1. In stage *k*, we make the optimal polling order of ONUs in $\phi(k)$, which is a *k*-element subset of all ONUs. The total number of $\phi(k)$ is C^k_N.

Assume $opt(\phi(k))$ and $V(\phi(k))$ as the optimal polling order of all ONUs in $\phi(k)$ and $\sum_{n=1}^{k} S_{[n]}$ under the order $opt(\phi(k))$, respectively. In Lines 6–8, for $k=1$, the initial condition can be expressed as follows:

$$V(\{i\}) = |d_i|, \quad \forall i (1 \le i \le N). \tag{19}$$

As there is only one ONU to poll, $opt(\{i\}) = \{i\}$. In Lines 10–16, we traverse all elements in $\phi(k)$ and then use $opt(\phi(k-1))$ and $V(\phi(k-1))$ of stage $k-1$ to get $opt(\phi(k))$ and $V(\phi(k))$ for $\phi(k)$. The recursion is

$$V(\phi(k)) = \min_{j \in \phi(k)} (V(\phi(k)/\{j\}) + |pos(\phi(k)/\{j\}) + d_j|) \tag{20}$$

where $\phi(k)/\{j\}$ and $pos(\phi(k)/\{j\})$ denote the $(k-1)$-element subset of $\phi(k)$ without ONU *j* and the difference between the upstream and the downstream transmission end time of the last granted ONU in $\phi(k)/\{j\}$, respectively. $pos(\phi(k))$ is computed as follows:

$$pos(\phi(k)) = \sum_{i \in \phi(k)} d_i \tag{21}$$

$pos(\phi(k))$ is only related to $\phi(k)$ because d_i is fixed by the granted bandwidth of ONU *i* in Eq. (17).

Found $V(\phi(k))$ with $J_{min} \in \phi(k)$ as the last polled ONU, the optimal polling order is the combination of $opt(\phi(k)/\{J_{min}\})$ and J_{min}.

In our algorithm, the optimal decision can be made by the results in the former stage. Therefore, we can obtain $opt(\mathbf{N})$ and $V(\mathbf{N})$ by the iteration using the dynamic programming algorithm.

Algorithm 1. Dynamic programming algorithm.

1: **for** $i = 1 \to N$ **do**
2: Compute d_i for ONU *i* according to Eq. (17).
3: **end for**
4: **for** $k = 1 \to N$ **do**
5: List all *k*-element subsets of all ONUs, as $\Phi(k)$
6: **if** $k == 1$ **then**
7: Calculate each $V(\phi(1))$ according to Eq. (19)
8: $opt(\phi(1)) = \{j\}, j \in \phi(1)$
9: **else**
10: **for** $\phi(k) \in \Phi(k)$ **do**
11: **for** ONU $j \in \phi(k)$ **do**
12: $v(\phi(k), j) = V(\phi(k)/\{j\}) + |pos(\phi(k)/\{j\}) + d_j|$

13: **end for**
14: $V(\phi(k)) \leftarrow \min_{j \in \phi(k)}(v(\phi(k), j))$,
15: $J_{min} \leftarrow j$ in $\min_{j \in \phi(k)}(v(\phi(k), j))$
16: $opt(\phi(k)) \leftarrow [opt(\phi(k)/\{J_{min}\}), J_{min}]$
17: **end for**
18: **end if**
19: **end for**

Complexity analysis: The complexity of Lines 1–3 is $O(N)$. The complexity of the "**for**" loop in Line 4 is $O(N)$. In stage k, the total number of different $\phi(k)$ is C_N^k. Based on Eq. (20), the comparison number to determine each $V(\phi(k))$ is k. So the total comparison number in stage k is $k \times C_N^k$. For the whole problem with N stages, the total comparison number N_{cmp} can be computed as follows:

$$N_{cmp} = \sum_{k=1}^{N} (k \cdot C_N^k) = N \cdot 2^{N-1} \tag{22}$$

The total complexity of Lines 4–19 is $O(N \cdot 2^N)$. Hence, the iterative algorithm has the computational complexity of $O(N \cdot 2^N)$.

4.2. Most fit first algorithm

The dynamic-programming-based algorithm incurs a high complexity to achieve optimality. Therefore, we propose a heuristic algorithm to compute a near-optimal solution with high computation efficiency. We introduce Theorem IV.1 first.

Theorem IV.1. *Given a polling order σ, exchanging the order of ONU $[n]$ and ONU $[n+1]$ in σ results in a new polling order σ'. If $S_{[n]}{}'$ under σ' is less than $S_{[n]}$ under σ, the total energy consumption by σ' is less than that by σ.*

Proof. As illustrated in Fig. 4, exchanging the order of ONU $[n]$ and ONU $[n+1]$ in σ only affects the value of $S_{[n]}$ while the values of other $S_{[j]}$ $(1 \le j < N$ and $j \ne n)$ remain the same. It can be shown as follows:

$$S_{[j]} = S'_{[j]}, \quad 1 \le j < N \text{ and } j \ne n \tag{23}$$

From Eq. (15), the total active time of all ONUs is only affected by $\sum_{n=1}^{N} S_{[n]}$. As $S'_{[n]} < S_{[n]}$, $\sum_{n=1}^{N} S'_{[n]}$ under σ' is less than $\sum_{n=1}^{N} S_{[n]}$ under σ. Hence, the polling order σ' achieves better performance in energy efficiency than σ. □

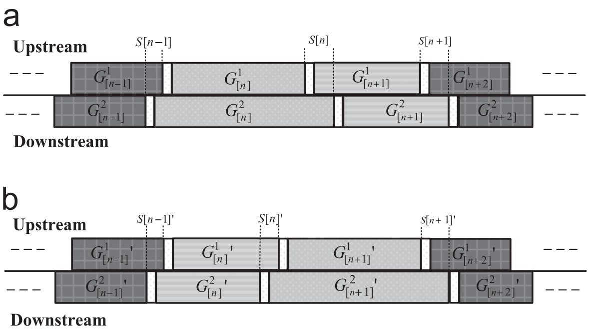

Fig. 4. Illustration of Theorem IV.1: (a) polling order σ and (b) polling order σ'.

Based on Theorem IV.1, we devise the most fit first algorithm (MFFA) in which the OLT always first polls the ONU which makes the difference between upstream and downstream transmission end times least. Though the proposed algorithm only obtains the local optimal solution, the result is not far from that of the optimal solution because the value of N is small in the actual EPON systems. Algorithm 2 details the most fit first algorithm.

Assume $heu(\mathbf{N})$ as the polling order made by the MFFA. In Lines 5–9, we pick the ONU j whose absolute difference between upstream and downstream granted transmission times $|d_j|$ is the least among all the ONUs in $\mathbf{N}$. As ONU j is the first one to be polled, $d_{[1]} = d_j$. Denote $s_{[n]}$ as the difference between the upstream and downstream transmission end times of the nth polled ONU. We can find $s_{[1]} = d_j$ and $S_{[1]} = |s_{[1]}|$. After ONU j is picked as the first polled ONU, the OLT removes ONU j from $\mathbf{N}$ and get the subset $\phi(N-1)$ in which we pick the second polled ONU.

In the decision process of nth polled ONU (Lines 11–14), we traverse all ONUs in $\phi(N-n+1)$ and pick the most fit ONU j to make $S_{[n]}$ least. Then we insert the ONU j at the end of $heu(\mathbf{N})$ and record the values of $d_{[n]}$, $s_{[n]}$ and $S_{[n]}$, as shown in Lines 12 and 13. $\phi(N-n)$ is updated by removing the selected ONU j from $\phi(N-n+1)$. When n increases to N, we can get the final polling order $heu(\mathbf{N})$ made by MFFA.

Algorithm 2. Most fit first algorithms

1:: **for** $i = 1 \rightarrow N$ **do**
2: Compute d_i for ONU i according to Eq. (17).
3: **end for**
4: **for** $n = 1 \rightarrow N$ **do**
5: **if** $n == 1$ **then**
6: Traverse all ONUs in $\mathbf{N}$ to find ONU j which satisfies $|d_j| = \min_{k \in \mathbf{N}} |d_k|$
7: Denote $heu(\mathbf{N})$ as constructed polling order: $heu(\mathbf{N}) \leftarrow \{j\}$,
8: $d_{[1]} \leftarrow d_j$, $s_{[1]} \leftarrow d_j$, $S_{[1]} \leftarrow |s_{[1]}|$
9: Remove ONU j from $\mathbf{N}$: $\phi(N-1) \leftarrow \mathbf{N}/\{j\}$
10: **else**
11: Traverse all ONUs in $\phi(N-n+1)$ to find ONU j which satisfies $|d_j + s_{[n-1]}| = \min_{k \in \phi(N-n+1)} |d_k + s_{[n-1]}|$
12: $heu(\mathbf{N}) \leftarrow [heu(\mathbf{N}), \{j\}]$,
13: $d_{[n]} \leftarrow d_j$, $s_{[n]} \leftarrow d_j + s_{[n-1]}$, $S_{[n]} \leftarrow |s_{[n]}|$
14: Remove ONU j from $\phi(N-n+1)$: $\phi(N-n) \leftarrow \phi(N-n+1)/\{j\}$
15: **end if**
16: **end for**

Complexity analysis: The complexity of Lines 1–3 is $O(N)$. The complexity of the "**for**" loop in Line 4 is $O(N)$. And the comparison number of the traversal in Line 11 is $N-n+1$. The total complexity of Lines 4–16 is $O(N^2)$. Hence, the most fit first algorithm has the computational complexity of $O(N^2)$.

The example of six ONUs as shown in Fig. 5 illustrates the most fit first algorithm. Fig. 5(a) shows the granted bandwidth of all ONUs. Fig. 5(b) shows the constructed polling order by our proposed algorithm. First, we compute the differences between upstream and downstream granted bandwidths of all ONUs and find that ONU 5 has the least difference. Then, the OLT polls ONU 5 first and then find the next ONU to be polled. We traverse the left ONUs and find that polling ONU 4 after ONU 5 can make

a

ONU (k)	1	2	3	4	5	6
Up Bandwidth (G_k^1)	2000 bytes	2000 bytes	4000 bytes	5000 bytes	2000 bytes	5000 bytes
Down Bandwidth(G_k^2)	5000 bytes	6000 bytes	2000 bytes	4000 bytes	3000 bytes	1000 bytes

b

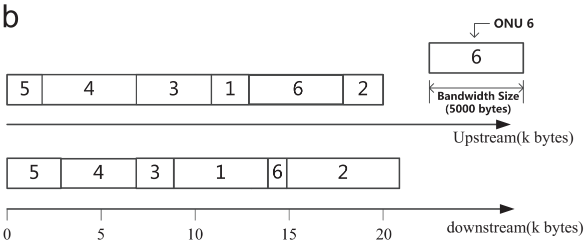

Fig. 5. Illustration of MFFA: (a) granted bandwidth and (b) scheduling result.

$S_{[2]}$ least. Based on the same principle, the OLT can decide the next polled ONU with the information of the previously polled ONU. Then, the heuristic polling order can be constructed.

5. Extension for special case

In this section, we introduce some extensions to improve the scheduling results for special cases (light load case and asymmetric load case). After constructing polling orders, the OLT can adjust the transmission windows with the extensions to obtain better energy efficiency.

5.1. Light load case

In the two proposed algorithms, the polling cycle length corresponds to the sum of the granted bandwidth allocated to all ONUs. If the network load is light, the average ONU requested bandwidth is low. Based on the grant sizing policy, the OLT allocates requested bandwidth to each ONU, which results in a shorter polling cycle than the pre-defined polling cycle T_{CYC}. And during the shorter polling cycle, the requested bandwidth of each ONU becomes much less. The vicious circle makes the ONUs remain idle only for a shorter period and switch between sleep/doze mode and active mode more frequently, resulting in significant energy waste in the state transition and the control message exchange.

Therefore, we set T_{cycmin} as the threshold of the polling cycle length. T_{cycmin} is predefined by the OLT before the polling cycle begins:

$$T_{cycmin} = \mu T_{CYC} \tag{24}$$

where μ is the weight of T_{CYC}. $\mu \le 1$.

If the current polling cycle length is shorter than T_{cycmin}, the OLT makes all ONUs remain idle until the polling cycle grows to T_{cycmin}. Under different QoS requirements, the OLT can change the value of μ to alter the threshold. With a larger threshold, the EPON can achieve better energy efficiency, but obtain longer delay. On the other hand, with a smaller threshold, the EPON gets better network performance, but consumes more energy. However, under heavy load, if T_{cycmin} is not enough to transmit the total granted bandwidth, the extension has no impact on the original algorithms.

5.2. Asymmetric load case

In reality, the network traffic may be asymmetric. The downstream bandwidth requests could be much larger than the upstream bandwidth requests. In the offline-based scheduling policy, the polling cycle length is affected only by the total granted bandwidth in the heavier-load direction. In our proposed algorithms, we do not allow any idle time between two adjacent transmission windows in both directions. In the lighter-load direction, the granted transmission window of one ONU may be allocated a long time before its granted transmission window in the heavier-load direction, which makes the ONU stay in active mode for a large time.

Therefore, in the lighter-load direction, we insert some idle time between two adjacent ONU transmission windows to save energy. Note that the OLT always avoids idle time in the heavier-load direction, this extension barely affects the network delay because of the fixed polling cycle length. Denote the total granted bandwidth in one direction as $Total_{\ell}$. Assume α and β as the lighter-load direction and heavier-load direction, respectively. $Total_{\alpha} \le Total_{\beta}$, so Eq. (12) can be released to

$$\begin{cases} T_{[n]}^{\alpha,s} \le T_{[n-1]}^{\alpha,e} + T_g, & \forall n (1 \le n \le N); \\ T_{[n]}^{\beta,s} = T_{[n-1]}^{\beta,e} + T_g, & \forall n (1 \le n \le N). \end{cases} \tag{25}$$

First, the OLT constructs the polling order of all ONUs based on the proposed grant scheduling algorithms. Then, we check if some idle time can be inserted between two adjacent transmission windows in the lighter-load direction. The condition to insert the idle time between nth and $(n+1)$-th polled ONU can be expressed as follows:

$$T_{[n]}^{\alpha,e} \le T_{[n]}^{\beta,e} \quad \text{and} \quad G_{[k]}^{\alpha} < G_{[k]}^{\beta}, \quad \forall k (n < k \le N). \tag{26}$$

Eq. (26) ensures that inserting the idle time between two adjacent ONU transmission windows can decrease the total active time of all the ONUs polled later. Finally, we compute the value of the idle time. We denote $Idle_{[n]}^{\alpha}$ as the idle time after the n-th polled ONU transmission window

a

ONU (k)	1	2	3	4	5	6
Up Bandwidth (G_k^1)	2000 bytes	2000 bytes	3000 bytes	1000 bytes	2000 bytes	5000 bytes
Down Bandwidth(G_k^1)	7000 bytes	6000 bytes	2000 bytes	5000 bytes	4000 bytes	1000 bytes

b

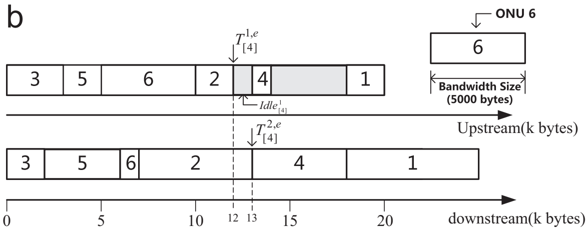

Fig. 6. Illustration of the extension for asymmetric load case: (a) granted bandwidth and (b) scheduling result.

in direction α. It can be computed as follows:

$$Idle_{[n]}^{\alpha} = T_{[n]}^{\beta,e} - T_{[n]}^{\alpha,e}. \tag{27}$$

Taking the example of a six-ONUs network, Fig. 6 illustrates the proposed extension. The total upstream bandwidth is much less than the total downstream bandwidth. So the idle time can be inserted in the upstream channel. After scheduling ONU 2, the upstream transmission end time $T_{[4]}^{1,e}$ is 12 and $T_{[4]}^{2,e}$ is 13. Further, the upstream granted bandwidth of the unassigned ONUs (ONU 4 and ONU 1) is less than their downstream granted bandwidth. So the idle time can be inserted after ONU 2. And the computed idle time is $13-12=1$. The last polled ONU 1 can also be polled with the extension.

6. Performance evaluation

6.1. Simulation setup

In this section, we investigate the performance of the different DBA algorithms we presented. We simulated an EPON system with a channel capacity, $R=1$ Gb/s and N ONUs ($N=8$ or 16). We assume that the distance between the OLT and the ONUs is 20 km. The guard time between two transmission windows is set to be 1 μs. To reflect the property of the real Internet traffic, we generate self-similar traffic by aggregating multiple sub-streams. Each sub-stream consists of alternating Pareto-distributed ON/OFF periods, with a Hurst parameter of 0.8. The packet sizes are uniformly distributed between 64 and 1518 bytes. The MPCP control message overhead is set to be 64 bytes. Furthermore, in order to combat the extreme uncertainty of self-similar traces and deliver conclusive results, the outcomes of multiple repeated simulation runs are averaged for each result. We assume that the total network load is evenly distributed amongst all ONUs and the ONUs are equally weighted. We also assume varying initial polling cycles (1 ms or 2 ms) to exploit the performance of different DBA algorithms.

In our work, the considered performance metrics include average packet delay and ONU average power, which are the global average value of all ONUs. Packet delay is defined as the time period from the instant of packet generation at an ONU to the complete delivery of the packet to the OLT. ONU power is defined as the energy consumption per unit of time.

6.2. Comparison of different algorithms

First, we simulate the EPON system with symmetric load, which means the total upstream load is equal to the total downstream load. We compare the performances of different algorithms in terms of the packet delay and the ONU power. Fig. 7 presents the performance comparison of the five algorithms. The 'UCS' represents the algorithm proposed in [24]. The 'UCS' only considers the upstream bandwidth requests and allocates active slots to all ONUs. Specifically, unlike the original 'UCS' in which a fixed active slot is allocated, we implement the 'UCS' such that a

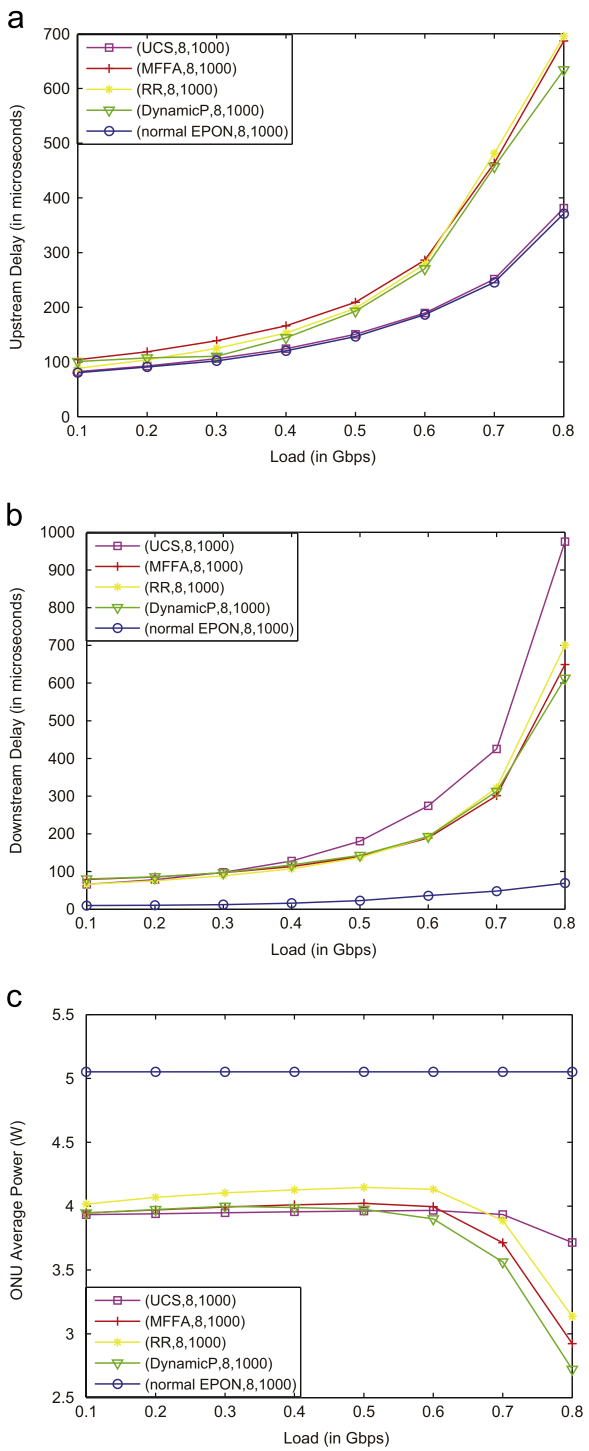

Fig. 7. Performance comparison of different algorithms at symmetric load: (a) upstream packet delay, (b) downstream packet delay and (c) ONU average power.

dynamic active slot with the excess bandwidth distribution is allocated to the ONUs. Only during the allocated active slot of one ONU, the OLT sends the downstream traffic to it. The excess downstream traffic has to be buffered at the OLT and waits for the next active slot.

The ‘RR’ polls all ONUs in a Round Robin manner under our proposed MAC control scheme and decides the allocated transmission windows with the excess bandwidth distribution in both directions. We also compare our algorithms (MFFA and DynamicP) with the conventional EPON which has no energy saving function.

Fig. 7(a) and (b) shows the average upstream and downstream packet delays versus the downstream aggregate load, respectively. The number of ONUs is 8 because of the high computational complexity of the dynamic programming algorithms. We can find that the ‘UCS’ achieves almost the best performance in upstream delay and the worst performance in downstream delay. The OLT in the ‘UCS’ allocates the transmission windows only with the consideration of the upstream bandwidth requests, which makes that unallocated downstream packets are buffered in the OLT. Though the ‘MFFA’ and ‘DynamicP’ may change the ONU polling order in each polling cycle, the impact on average delay would be negligible for a long time period. It is because that the transmission slot intervals of each ONU under the two algorithms would be randomly distributed with a mean value of one cycle length, which is almost the same with that under the ‘RR’ algorithm. Furthermore, they all consider the bandwidth requests in both directions so the upstream packets may experience more delay than in the other two algorithms. The conventional EPON without energy saving function always has the best performance.

Fig. 7(c) shows the ONU average power versus the downstream traffic load. The normal EPON keeps all ONUs active and has the largest ONU average power. And, for the other four energy-efficient algorithms, the ONU average powers at light load are almost constant and are larger than those at heavy load. At light load, the requested bandwidth is small and the cycle length is short, which makes the idle time for each ONU short. So the ONUs enter into doze mode instead of sleep mode due to the large overhead time from sleep mode to active mode. At heavy load, the polling cycle length becomes larger so that ONUs can enter into sleep mode to reduce more energy consumption. To avoid the light-load penalty, we introduce the extension in Section 5, which will be discussed later. The dynamic programming algorithm, which is represented by the ‘DynamicP’, outperforms the other algorithms because it gets the optimal polling order to save energy. The ‘MFFA’ slightly underperforms the dynamic programming algorithm and outperforms the ‘RR’ algorithm. At heavy load, the ‘UCS’ has a higher average ONU power than other algorithms because the polling cycle length by the ‘UCS’ is less, which leads ONUs polled by the ‘UCS’ to have less idle time during the polling cycle.

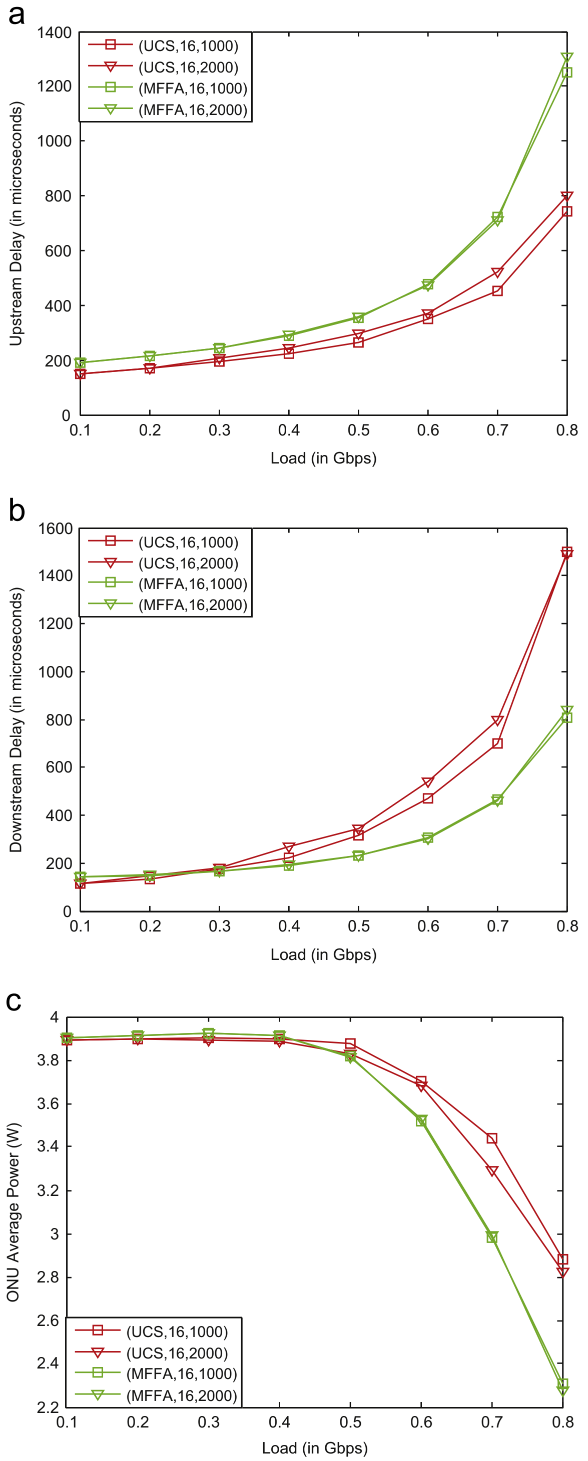

Fig. 8. Impact of cycle length at symmetric load: (a) upstream packet delay, (b) downstream packet delay and (c) ONU average power.

6.3. Impact of ONU number and cycle length

To exploit the impact of the ONU number and the cycle length, we compare the ‘UCS’ and ‘MFFA’ with different ONU numbers (8 or 16) and initial cycle lengths (1 ms or 2 ms). From Figs. 8 and 9, we can find that the initial cycle length has little impact on the packet delay and the ONU average power, and the algorithm with a larger ONU number obtains larger packet delay and lower ONU average power. The EPON system with a larger ONU number allocates less active time for each ONU and let ONUs remain idle for a longer period. The incoming packets for each ONU experience more delay in the buffer to be transmitted. And during the waiting time, the ONU

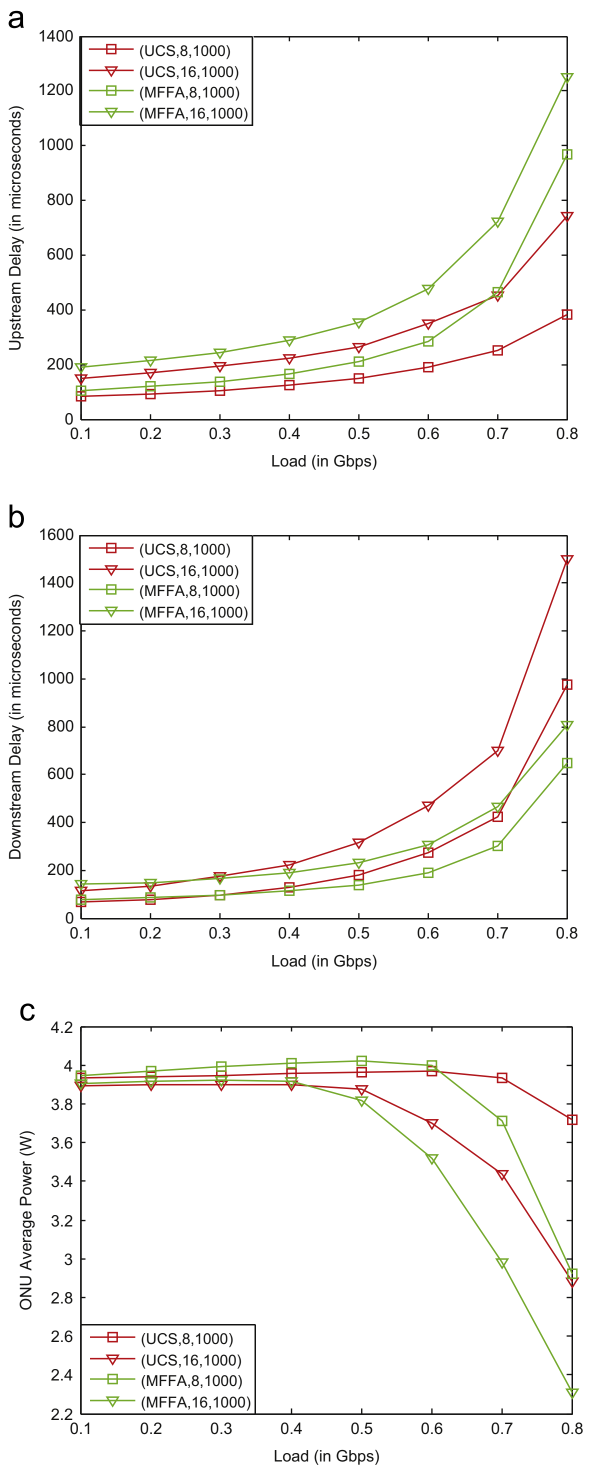

Fig. 9. Impact of ONU number at symmetric load: (a) upstream packet delay, (b) downstream packet delay and (c) ONU average power.

can keep in sleep/doze mode to reduce more energy consumption. On the other hand, a larger initial cycle length does not affect the performance of the EPON system substantially. Although the smaller cycle length results in less available bandwidth to be granted for each ONU, it polls ONUs more frequently, which can offset the effect of less bandwidth availability.

6.4. Extension for light load case

In Fig. 7, the average ONU power at light load is higher than that at heavy load. We introduce an extension for our algorithms to reduce the energy consumption at light load. Fig. 10 shows the performance comparison among the algorithms with and without the extension. We set the minimal polling cycle length threshold as $0.3T_{CYC}$ (T_{CYC} is 1 ms here), which prevents frequent ONU mode transition between active and sleep/doze modes.

In Fig. 10(a), the algorithms with the minimal polling cycle threshold get larger upstream packet delay than the algorithms without the threshold at light load. The difference of the upstream packet delays is about $0.3T_{CYC}$ according to the minimal polling cycle length threshold. When the load is 0.8, the packet delay obtained by the algorithms with the threshold approaches to that obtained by the algorithms without the threshold. Fig. 10(b) presents the performance comparison on the downstream packet delay. Similar to the upstream packet delays, the downstream delays obtained by the algorithms with the threshold are also larger than that obtained by the algorithms without the threshold. The threshold has much more effect on the 'UCS' algorithm because the 'UCS' only considers the upstream bandwidth requests.

At light load, the ONU average power is significantly reduced with the algorithms with the threshold, as shown in Fig. 10(c). The performance achieved by the most fit first algorithm is almost the same as that achieved by the dynamic programming algorithm. Although the 'UCS' with the threshold slightly outperforms our proposed algorithms on the ONU average power, the downstream packet delay degeneration by the minimal polling cycle threshold is significant. Without much network performance degeneration, our proposed algorithms with the minimal polling cycle threshold can significantly reduce the energy consumption of ONUs.

6.5. Extension for asymmetric load case

When the upstream and downstream loads are asymmetric, some idle time is allowed between two transmission windows in the lighter-load direction to reduce the ONU active time and save energy, as noted in Section 5. In this simulation, we set that the downstream aggregate load is twice of the upstream aggregate load.

Fig. 11 shows the performance comparison at asymmetric load. We apply the extension for the light load case to each algorithm. As the 'UCS' schedules ONUs by the upstream bandwidth requests only, inserting idle time between ONU transmission windows is not possible for it. Fig. 11(a) shows the average upstream packet delay versus the downstream aggregate load. All energy efficient algorithms have almost the same upstream packet delay at light load, which is mainly related to the minimal polling cycle threshold. With the load increased, the packet delay of the 'UCS' remains almost constant while the packet delays of the other algorithms increase. The algorithms

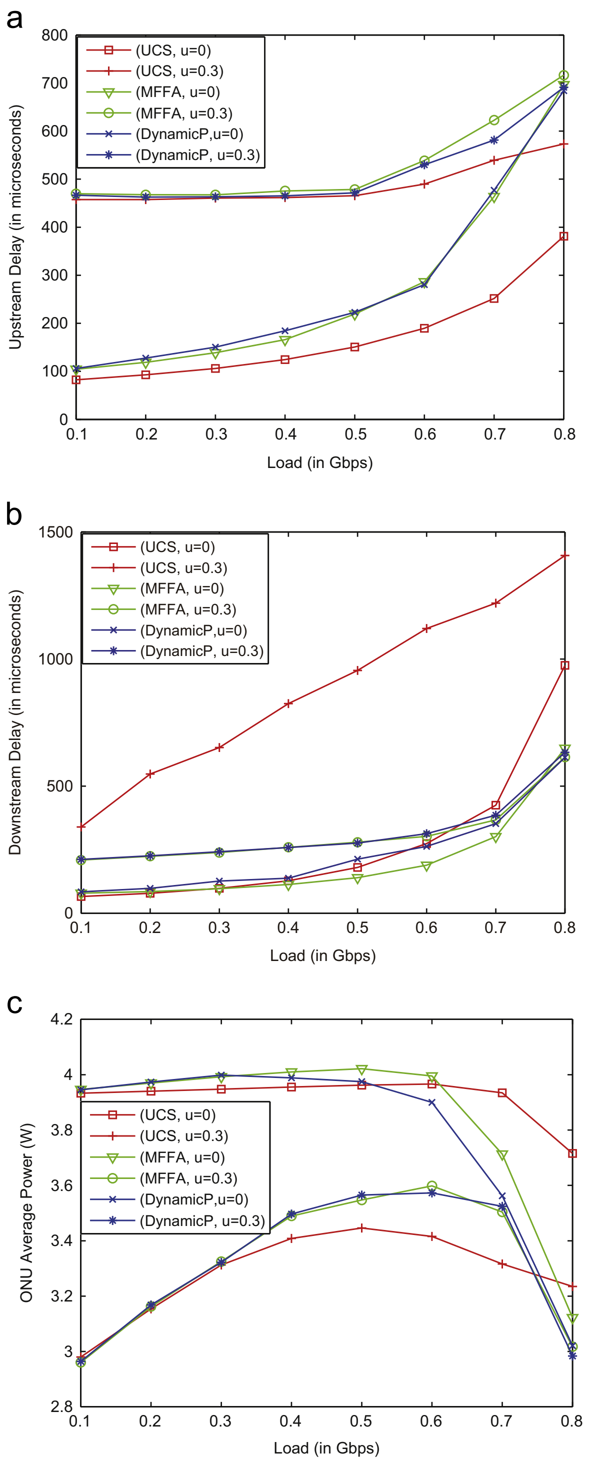

Fig. 10. Extension for light load case: (a) upstream packet delay, (b) downstream packet delay and (c) ONU average power.

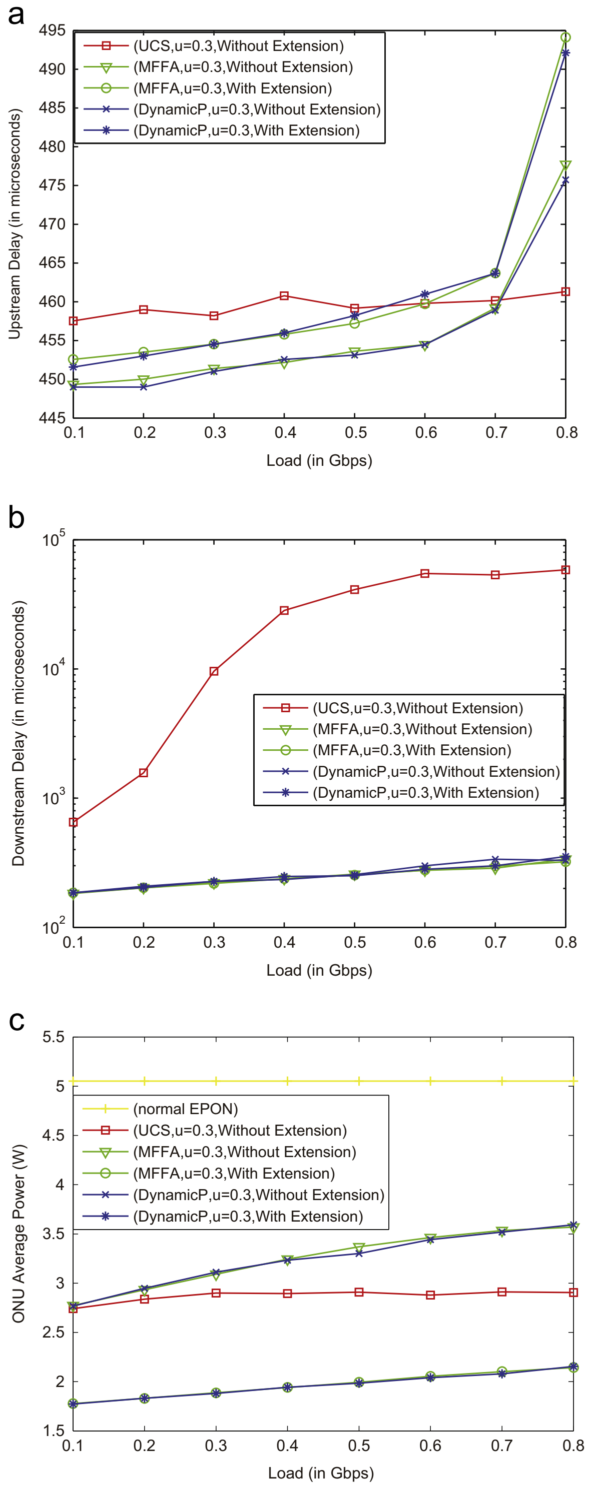

Fig. 11. Extension for asymmetric load case: (a) upstream packet delay, (b) downstream packet delay and (c) ONU average power.

without the extension outperform the algorithms with the extension a little in terms of the upstream delay. This is because some idle time would be inserted between two ONU transmission windows, resulting in some delay. In Fig. 11(b), the downstream packet delay of the 'UCS' is much larger than that of the other algorithms at asymmetric load. As the total downstream load is larger than the total upstream load, the 'UCS' makes the downstream packets to wait for a much longer time for transmission when it only considers the upstream bandwidth requests.

Our proposed algorithms with or without the extension perform almost the same. The result shows that applying the extension to our proposed algorithms has little impact on the packet delay at asymmetric load.

From Fig. 11(c), we can find that the proposed extension for the asymmetric load case can significantly reduce the average ONU power. Introducing some idle time to the lighter-load direction makes the upstream and downstream transmission windows of ONUs overlap more and can reduce the active time of them. In addition, the most fit first algorithm and the dynamic programming algorithm obtain the same performance due to the small ONU number.

7. Conclusion

In this paper, we designed an energy efficient DBA framework, including a MAC control scheme, a grant sizing policy and two grant scheduling algorithms. The MAC control scheme periodically puts ONUs into sleep/doze mode to reduce energy consumption. Based on the Multi-Point Control Protocol, it allows the ONUs with the sleep function to coexist with the ONUs in legacy EPONs. Our problem formulation for the energy efficient grant scheduling can precisely capture the key optimization target of the system, and is simplified for high computation efficiency without understating the problem. To achieve high energy efficiency, we introduced two algorithms with tolerable network performance degradation. While the dynamic-programming-based algorithm achieves the optimal result, we showed that the heuristic version can greatly reduce computation complexity and meanwhile achieves a reasonably good result. We also patched the algorithms by considering the extreme network scenarios (i.e., light load). Simulation results show that the proposed DBA framework can achieve significant energy saving especially with the extensions for special network cases.

Acknowledgments

This research is sponsored by the China NSFC under Contract no. 61071080, the China 973 program under Grants 2010CB328204, 2010CB328205, and the China 863 program.

References

[1] M. Gupta, S. Singh, Greening of the internet, in: Proceedings of ACM SIGCOMM'03, Karlsruhe, Germany, August 2003.
[2] J. Baliga, R. Ayre, K. Hinton, W.V. Sorin, R.S. Tucker, Energy consumption in optical IP networks, IEEE/OSA J. Lightw. Technol. 27 (13) (2009) 2391–2403.
[3] J. Kani, S. Shimazu, N. Yoshimoto, H. Hadama, Energy-efficient optical access networks: issues and technologies, IEEE Commun. Mag. 51 (2) (2013) S22–S26.
[4] A. Dhaini, P. Ho, G. Shen, B. Shihada, Energy efficiency in TDMA-based next-generation passive optical access networks, IEEE ACM Trans. Netw. PP (99) (2013) 11.
[5] P. Chowdhury, M. Tornatore, S. Sarkar, B. Mukherjee, Building a green wireless-optical broadband access network (WOBAN), IEEE/OSA J. Lightw. Technol. 28 (16) (2010) 2219–2229.
[6] J. Hsieh, T. Lee, Y. Kuo, Energy-efficient multi-polling scheme for wireless LANs, IEEE Trans. Wirel. Commun. 8 (3) (2009) 1532–1541.
[7] P. Lu, Y. Yuan, Z. Yang, Z. Zhu, On the performance analysis of energy-efficient upstream scheduling for hybrid fiber-coaxial networks with channel bonding, IEEE Commun. Lett. 17 (5) (2013) 1020–1023.
[8] K.J. Christensen, C. Gunaratne, B. Nordman, A.D. George, The next frontier for communications networks: power management, Comput. Commun. 27 (2004) 1758–1770.
[9] GPON Power Conservation, ITU-T G-Series Recommendations, Supplement 45(G.sup45), 2009.
[10] 10-Gigabit-Capable Passive Optical Networks XG-PON (G9873), ITU-T G.987.x-Series Recommendations, 2010.
[11] IEEE Standard for Service Interoperability in Ethernet Passive Optical Networks (SIEPON), IEEE Std 1904.1-2013, 2013.
[12] J. Mandin, EPON Powersaving Via Sleep Mode, IEEE 802.3av 10GEPON Task Force Meeting, September 2008.
[13] B. Skubic, D. Hood, Evaluation of ONU power saving modes for gigabit-capable passive optical networks, IEEE Netw. 25 (2) (2011) 20–24.
[14] A.R. Dhaini, P. Ho, G. Shen, Toward green next-generation passive optical networks, IEEE Commun. Mag. 49 (11) (2011) 94–101.
[15] S. Wong, L. Valcarenghi, S. Yen, D.R. Campelo, S. Yamashita, L. Kazovsky, Sleep mode for energy saving PONs: advantages and drawbacks, in: Proceedings of IEEE GLOBECOM'09, Hawaii, USA, December 2009.
[16] J. Zheng, H. Mouftah, Media access control for Ethernet passive optical networks: an overview, IEEE Commun. Mag. 43 (2) (2005) 145–150.
[17] M. McGarry, M. Reisslein, C. Colbourn, M. Maier, F. Aurzada, M. Scheutzow, Just-in-time scheduling for multichannel EPONs, IEEE/OSA J. Lightw. Technol. 26 (10) (2008) 1204–1216.
[18] G. Kramer, B. Mukherjee, G. Pesavento, IPACT a dynamic protocol for an Ethernet PON (EPON), IEEE Commun. Mag. 40 (2) (2002) 74–80.
[19] M. McGarry, M. Reisslein, F. Aurzada, M. Scheutzow, Shortest propagation delay (SPD) first scheduling for EPONs with heterogeneous propagation delays, IEEE J. Sel. Areas Commun. 28 (6) (2010) 849–862.
[20] M. McGarry, M. Reisslein, Investigation of the DBA algorithm design space for EPONs, IEEE/OSA J. Lightw. Technol. 30 (14) (2012) 2271–2280.
[21] C. Assi, Y. Ye, S. Dixit, M. Ali, Dynamic bandwidth allocation for quality-of-service over ethernet PONs, IEEE J. Sel. Areas Commun. 21 (9) (2003) 1467–1477.
[22] X. Bai, C. Assi, A. Shami, On the fairness of dynamic bandwidth allocation schemes in ethernet passive optical networks, Comput. Commun. 29 (11) (2006) 2125–2135.
[23] J. Ferguson, M. Reisslein, M. McGarry, Online excess bandwidth distribution for Ethernet passive optical networks, J. Opt. Netw. 8 (4) (2009) 358–369.
[24] Y. Yan, S.-W. Wong, L. Valcarenghi, S.-H. Yen, D.R. Campelo, S. Yamashita, L. Kazovsky, L. Dittmann, Energy management mechanism for Ethernet Passive Optical Networks (EPONs), in: Proceedings of IEEE ICC'10, Cape Town, South Africa, May 2010.
[25] L. Zhang, Y. Liu, L. Guo, X. Gong, Energy-saving scheme based on downstream packet scheduling in ethernet passive optical networks, Opt. Fiber Technol. 19 (2) (2013) 169–178.
[26] M. Dias, E. Wong, Sleep/doze controlled dynamic bandwidth allocation algorithms for energy-efficient passive optical networks, Opt. Express 21 (8) (2013) 9931–9946.
[27] S. Newaz, A. Cuevas, G. Lee, N. Crespi, J. Choi, Adaptive delay-aware energy efficient TDM-PON, Comput. Netw. 57 (7) (2013) 1577–1596.
[28] J. Zhang, N. Ansari, Toward energy-efficient 1G-EPON and 10G-EPON with sleep-aware MAC control and scheduling, IEEE Commun. Mag. 49 (2) (2011) 33–88.
[29] D. Pham Van, L. Valcarenghi, M. Chincoli, P. Castoldi, Experimental evaluation of an energy efficient TDMA PON, in: IEEE International Conference on Communications (ICC), 2013, pp. 3868–3872.
[30] D. Pham Van, L. Valcarenghi, M. Chincoli, P. Castoldi, Experimental evaluation of a sleep-aware dynamic bandwidth allocation in a multi-ONU 10G-EPON testbed, Opt. Switch. Netw. Available online 25 January 2014, ISSN 1573-4277, http://dx.doi.org/10.1016/j.osn.2014.01.006.
[31] IEEE Standard for Ethernet, Section 5, IEEE Std 802.3-2012 (Revision to IEEE Std802.3-2008), 2012.